Improving the Accessibility of Mobile OCR Apps Via Interactive Modalities

MICHAEL CUTTER and ROBERTO MANDUCHI, University of California, Santa Cruz

We describe two experiments with a system designed to facilitate the use of mobile optical character recognition (OCR) by blind people. This system, implemented as an iOS app, enables two interaction modalities (autoshot and guidance). In the first study, augmented reality fiducials were used to track a smartphone's camera, whereas in the second study, the text area extent was detected using a dedicated text spotting and text line detection algorithm. Although the guidance modality was expected to be superior in terms of faster text access, this was shown to be true only when some conditions (involving the user interface and text detection modules) are met. Both studies also showed that our participants, after experimenting with the autoshot or guidance modality, appeared to have improved their skill at taking OCR-readable pictures even without use of such interaction modalities.

CCS Concepts: • **Human-centered computing** → **Accessibility technologies**; *Empirical studies in ubiquitous and mobile computing*;

Additional Key Words and Phrases: OCR, accessibility, mobile devices, blindness

ACM Reference format:
Michael Cutter and Roberto Manduchi. 2017. Improving the Accessibility of Mobile OCR Apps Via Interactive Modalities. *ACM Trans. Access. Comput.* 10, 4, Article 11 (August 2017), 27 pages.
https://doi.org/10.1145/3075300

1 INTRODUCTION

Several optical character recognition (OCR) mobile phone apps specifically designed for blind users have recently appeared on the market (Holton 2016). These apps enable on-the-go access to printed documents such as restaurant menus, bills, signs on a door or on a wall, and class handouts. The processing power of modern smartphones, the excellent imaging characteristics and high resolution of smartphone cameras, and the maturity of OCR algorithms enable mobile OCR reading at a quality that is becoming comparable to that of traditional flatbed scanners (Coughlan and Manduchi 2013).

Research reported in this publication was supported by the National Eye Institute of the National Institutes of Health under award 1R21EY025077-01. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.
Authors' address: M. Cutter and R. Manduchi, University of California, Santa Cruz, 1156 High Street, MS: SOE3, Santa Cruz, CA 95064; emails: {mcutter, manduchi}@soe.ucsc.edu.

https://doi.org/10.1145/3075300

Yet a mobile OCR system that works perfectly for sighted people may be difficult for a blind individual to use. This is because OCR requires a well-framed image of text at good resolution to produce meaningful results. This is something that can be difficult to obtain without sight (Vázquez and Steinfeld 2012; Jayant et al. 2011). For example, a blind user of these apps may take pictures that are too far from the document, resulting in poor resolution, or too close, in which case the text may be cropped by the camera's viewport. Similarly, the smartphone may be mistakenly held sideways with respect to the document or oriented at the wrong angle. Lacking visual feedback, these situations may occur quite commonly, reducing the usability of this otherwise tremendously useful tool (Cutter and Manduchi 2015).

In fact, even sighted people sometimes have problems correctly framing a document with a camera. For example, banking apps that allow for check deposits using the smartphone camera (Burks et al. 2012) normally provide indications through visual means (e.g., showing fiducials on the smartphone's screen) that help the user correctly frame the check. Clearly, this problem is much more acute for blind persons, who are unable to see the smartphone's screen.

Some OCR apps provide accessibility tools that facilitate acquisition of well-framed, OCR-readable images by blind people. For example, Text Detective (available for iOS and Android) lets the user move the smartphone in front of the document while the camera continuously takes images at a relatively high rate; each image is analyzed by a fast text spotting algorithm, and as soon as the presence of readable text is detected, the image is sent on to OCR processing. In a similar fashion, Prizmo[1] and KNFB Reader[2] analyze the stream of images acquired by the camera in real time to detect whether all four edges of the document are visible. In addition, the system could produce nonvisual directions to help the blind user reorient or move the camera to take a better picture of the document. The "Field of view report" generated by the KNFB Reader app (which describes the position and orientation of the camera relative to the document) could be considered as a simple example of this modality. Prizmo (when run with VoiceOver enabled) provides a simple guidance mechanism, with directions such as "Up" or "Left" uttered by synthetic voice.

This article describes two experiments conducted with the purpose to study how blind people can use mobile OCR apps to access printed text and the extent to which system interaction can simplify this process. The two experiments were designed to validate similar hypotheses but with rather different apparatuses. Although the two studies were similar in purpose, they contributed different types of knowledge about the challenges of using mobile OCR without sight, and about the opportunities provided by computer vision algorithms coupled with an appropriate user interface.

Several hypotheses were tested in these experiments. We asked whether and in which situations a properly designed *guidance* modality could accelerate the process of acquiring an OCR-readable image of a document. We investigated whether experience with system interaction could actually help blind persons improve their proprioceptive skills, which are necessary for efficient use of a camera without sight. We also studied the effect of font size on one's ability to complete a proper image acquisition task. In addition, we took multiple measurements that shed light on various aspects of the process of accessing a document as mediated by a camera.

An earlier version of the material in Section 4 (Study 1) appeared in a prior conference paper (Cutter and Manduchi 2015). A preliminary experiment using a simpler version of the system described in Section 4, tested by eight blindfolded volunteers, was described in Cutter and Manduchi (2013).

[1] https://creaceed.com/prizmo.
[2] http://www.knfbreader.com.

2 RELATED WORK

Document scanners coupled with OCR and text-to-speech have been used successfully by many blind people to access printed text (Coughlan and Manduchi 2013). In recent years, several mobile OCR applications have been introduced to the market to enable quick text access "on the go." Three of the most popular apps are the KNFB Mobile Reader, Text Detective, and Prizmo. All three apps have mechanisms for automatic snapshot triggering (autoshot), which will be explained in Section 3.

Kane et al. (2013) developed a digital desk assistive environment that allowed blind people to interact with complex paper documents. This acquisition technology was based on a desktop camera, which captured a live stream of images. The largest contour in an image was assumed to identify the document's edges; the document was then processed by OCR.

The difficulty of taking good pictures without sight represents a hurdle not only for mobile OCR but also for other applications of camera-based information access, as well as for recreational photography. For example, Bigham et al. (2010b) used simple computer vision techniques along with crowdsourcing to help a blind user point a camera correctly at an object (e.g., to better identify it or get closer to it). Brady et al. (2013) analyzed the type of objects blind people take photos of in a crowdsourcing answer seeking scenario. Their analysis also included photo quality assessment. They found that 46% of the questions asked by their recent power users regarded reading some text. Taking good pictures of barcodes without sight is also difficult (Al-Khalifa 2008).

Zhong et al. (2013) developed a key-frame selection algorithm to be used in combination with a cloud-based visual search engine designed to help blind people identify objects. Experimental results showed that automatic key-frame selection from a video led to a higher success rate compared to when users themselves decided when to take a snapshot.

EasySnap and PortraitFramer are mobile applications developed by Jayant et al. (2011) that give feedback to a blind photographer about the scene light or about the presence and location in the picture of an object or a person. The use of real-time feedback to help a blind person document transit accessibility by taking pictures of the scene was studied by Vázquez and Steinfeld (2012). In this scenario, there was no clearly defined "target" (e.g., a face) that could be used to guide framing. Instead, a general-purpose saliency map was used to select a region of interest. Manduchi and Coughlan (2014) experimented with a feedback system designed to help a blind person reach a target (a color marker) using a camera phone. Accessing barcodes and QR codes via smartphones also presents difficulties for those without sight (Al-Khalifa 2008). A camera-based system for barcode access, equipped with a guidance mechanism that suggested how to move the camera to precisely center a detected barcode, was developed by Tekin and Coughlan (2010).

The process of taking a precisely framed picture of a document for OCR processing could potentially be facilitated by stitching together multiple pictures, each containing a partial view of the document, into a panoramic image (or mosaic) of the whole document, as suggested by Zandifar and Chahine (2002). A similar mechanism was used by Zhong et al. (2015) in their RegionSpeak system to facilitate exploration of a spatial layout.

Several systems proposed for text reading used dedicated hardware. Shilkrot et al. (2015) and, in separate work, Stearns et al. (2016) designed a finger-mounted camera that could be used to scan a text line. OrCam[3] is a wearable camera with a dedicated embedded computer that enables users to trigger OCR reading by pointing at the text with their index finger. Although these systems have shown good experimental results, we believe that commodity hardware such as a smartphone may be more attractive to potential users than a specialized assistive technology device.

[3] http://www.orcam.com.

A different approach to facilitating blind photography of a document is to use a stand to support the smartphone. Examples include the Samsung Optical Scan Stand[4] (which works only for the Galaxy Core Advance model), scanJig,[5] and Giraffe Reader.[6] Although effective, these systems require carrying one more accessory, which may limit their appeal.

Text access is possible even without OCR. The Optacon [Stein 1998], marketed by Telesensory Systems from 1971 to 1996, was a sensory substitution system that used an array of vibrating pins to convey the shape of individual letters, acquired by a camera operated by the user. Crowdsourcing (or friendsourcing) apps such as VizWiz [Bigham et al. 2010a], TapTapSee,[7] BeMyEyes,[8] and Aira[9] connect a blind user to a remote sighted helper who can read the text from an image or a video taken by the user's smartphone.

3 INTERACTION MODALITIES

We have identified three general interaction modalities that are employed in mobile OCR apps: manual, autoshot, and guidance.

Manual. This modality describes a mobile OCR system with basic interaction capabilities, such as the KNFB Reader in the "Manual picture" mode. The user moves the smartphone over the document (the camera pointing down) and takes a snapshot (by pressing a button or tapping on the phone's screen) when he or she thinks a good picture of the document could be obtained. The user can then listen to the OCR output produced via synthetic speech and evaluate if the text was correctly read. If parts of the text are missing, or are incorrectly interpreted by OCR, the user may need to start the process all over again.

We should note that the KNFB Reader has a modality ("Field of view report") by which the user can take an evaluation snapshot; the system then processes this image and communicates information about the quality of framing (e.g., "The right edge, top edge, and bottom edge of the page are visible, rotated 79 degrees counterclockwise.") The user, on hearing this information, may decided to move the smartphone to a better position (in the preceding example, move it to the left and rotate it clockwise) and take a new snapshot.

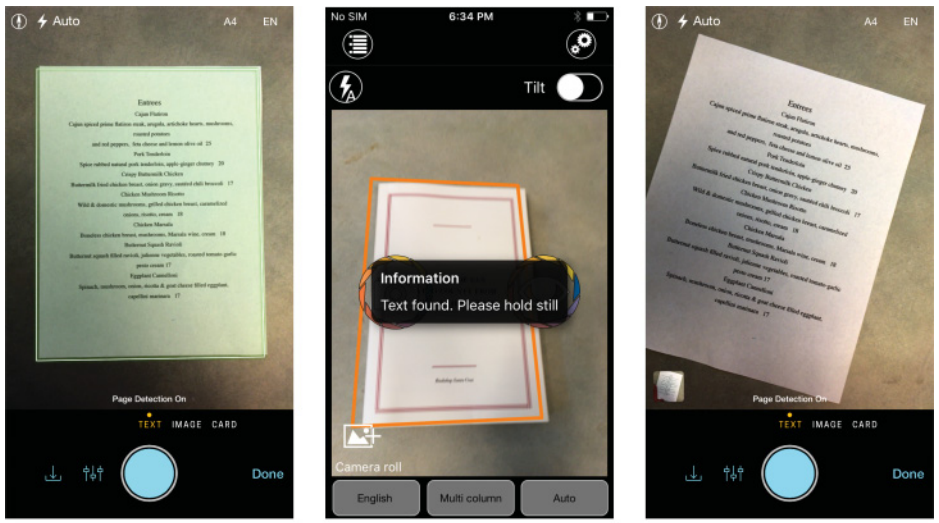
Autoshot. In this modality, the user is not required to trigger a snapshot manually: instead, it is the system that decides when to take an image to be processed by OCR. This capability is enabled by an algorithm that analyzes in real time the images continuously taken by the camera while the user is moving the smartphone over the document. These images need to be acquired and processed at an as high as possible frame rate; full resolution is typically not necessary. The algorithm detects when one or more consecutive frames in the sequence are *compliant*, triggering a high-resolution (hi-res) snapshot (possibly with the flash activated) to be sent to OCR. In this context, *compliance* is a generic term to indicate that an image is considered (by the system) to be correctly OCR readable (e.g., well framed, well lit, at good enough resolution). The main purpose of autoshot is to increase the likelihood of taking a correctly framed, OCR-readable snapshot, reducing the risk that the user may need to take another (or possibly multiple other) snapshots before satisfactory OCR reading.

Autoshot mechanisms have been used in mobile banking apps for making check deposits. Several existing mobile OCR systems designed for blind users also implement some form of autoshot. For example, Text Detective (Figure 1) takes a hi-res snapshot when the presence of text has been

[4] http://www.samsung.com/se/consumer/mobile-devices/accessories/others/EE-DI858BWEGWW.
[5] http://www.scanjig.com.
[6] http://www.giraffe-reader.com.
[7] http://www.taptapseeapp.com.
[8] http://www.bemyeyes.com.
[9] http://www.aira.io.

(a) (b)

Fig. 1. (a) Text spotting techniques can be used for triggering OCR processing on the image. (b) However, the presence of text is not by itself a good indicator that the document is readable, as text could extend outside the camera's viewport. Screenshots from Text Detective.



(a) (b) (c)

Fig. 2. Successful example of compliance detection by identifying an edge quadrilateral (Prizmo (a); KNFB Reader (b)). Note the detected quadrilateral, which is shown in color as superimposed on the image on the viewfinder. If not all four edges of the document are visible (c), no quadrilateral is detected, and the image is considered not compliant (even though the text could be fully read, as in the example shown in the figure).

detected in the image by a text spotter (similar in purpose to the algorithm used in our Study 2 system). Prizmo and KNFB Reader both have a mode ("Page detection" and "Automatic picture," respectively) that automatically triggers a hi-res snapshot when a brightness edge pattern forming a quadrilateral (e.g., the edges of a printed document) is detected (Figure 2).

These existing methods are not without their shortcomings. For example, Text Detective may decide to take a snapshot even when only part of the text in the image is visible (see Figure 1(b)). The edge-based approach of Prizmo and KNFB Reader may fail to detect the document's edges in case of low contrast (Figure 3(a)), or may take snapshots of a document with no text on it (see Figure 3(b) and (c)), or with text imaged at a resolution that is insufficient for OCR reading.

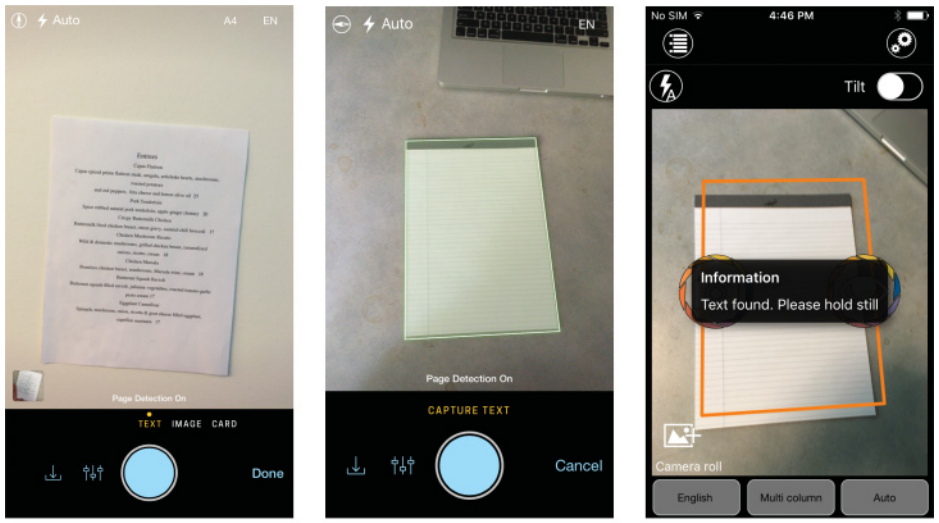


Fig. 3. Compliance detection by identification of a visible edge quadrilateral may fail if there is little contrast, such as on a white background (Prizmo (a)). It can also incorrectly trigger a snapshot in the absence of visible text (Prizmo (b); KNFB Reader (c)).

Guidance. This modality builds on the autoshot mode, with additional feedback provision from the system. Specifically, directions are produced via synthetic speech, which is meant to help the user move the camera to a position from which a compliant image could be taken. The hope is that this system feedback may facilitate the process of aiming the camera correctly and thus reduce the time to acquire an OCR-readable snapshot.

It is important to observe that both the horizontal and vertical components of the camera's location are important for image compliance. A picture taken from a camera that is too close to the document will likely fail to contain all text in the document. On the converse, if the camera is too far from the document, the smaller characters in the text may be imaged with a resolution that is insufficient for correct OCR reading.

To the best of our knowledge, Prizmo is the only mobile OCR system on the market that implements a form of guidance (when VoiceOver is activated). Directions are produced when a quadrilateral containing the edges of the document is detected (as in Prizmo's autoshot modality). Specifically, the system utters "Up" or "Down" when the area of the quadrilateral is larger/smaller than a threshold; "Left," "Right," "Back," or "Away" when the center of the quadrilateral is offset with respect to the image center; and "Ready to shoot" when the quadrilateral is well centered and of proper size. In addition, the system utters "No visible page" when no quadrilateral is detected and "Page detected" on detection. However, this is a simple and fairly effective mechanism that fails when the document is not fully framed by the image, as in this case the system cannot detect an edge quadrilateral (see Figure 2(c)). In our opinion, this is a serious shortcoming, as incorrect document framing due to horizontal camera offset or a camera too close to the document is a common situation that could benefit from corrective system feedback.

4 STUDY 1

Study 1 was conducted in spring 2013 with 12 blind participants. Its purpose was to verify whether the use of interactive modalities (autoshot or guidance) could increase the proficiency of a blind

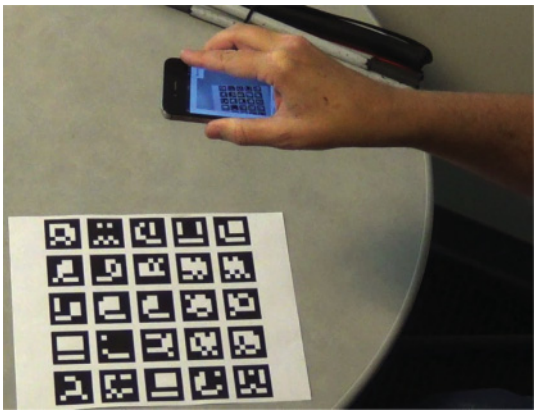


Fig. 4. A participant positioning the iPhone over the document printed with the ArUco fiducials in Study 1 (from Cutter and Manduchi (2015)).

user at capturing printed text with respect to the simpler manual modality. A secondary objective was to verify whether experience with these interactive modalities could help one learn to take good pictures of a document even without system interaction.

We created an iPhone app that implemented the manual, autoshot, and guidance modalities. Rather than containing text, the document used in this study had special augmented reality (AR) fiducials printed on it (Figure 4). A computer vision algorithm was used to process an image containing one or more such fiducials to measure location and orientation (collectively called *pose*) of the camera. Based on this information, the system estimates whether the image of the document would be compliant if the document actually were fully printed with standard-size text.

This quasi–Wizard of Oz approach allowed us to separate the technical difficulties of image compliance estimation from the human factors that pertain to holding a camera and taking a compliant picture. In addition, analysis of the recorded camera poses allowed us to obtain interesting information about the reasons for failure to take compliant pictures.

4.1 Method

4.1.1 Participants. Twelve blind participants (four females and eight males) were recruited through announcements on newsletters and word of mouth. The participants were between 18 and 65 years of age, with a median age of 53. Participants were divided into two groups (Group A and Group B, with six participants each) via random assignment.

All but one participant had at most some residual light perception. The participant who had some residual vision left had acuity of 20/3800 in one eye; the other eye had no vision (prosthetic). To remove any possibility that the little residual vision could bias results, this participant was blindfolded during the test. Seven participants (three in Group A and four in Group B) were regular iPhone users. Four participants (three in Group A and one in Group B) had tried mobile OCR systems before but were not regular users of this technology.

4.1.2 Apparatus. The app developed for this experiment ran on an iPhone 4S (with image resolution of 640×480 pixels). The autoshot and guidance modalities relied on a system that detected compliance from the camera pose, computed using AR fiducials printed on the document. In the context of this study, a *compliant* picture of a document is a picture that contains all of the text in the document at enough resolution that it can be read by OCR. More precisely, a picture of a

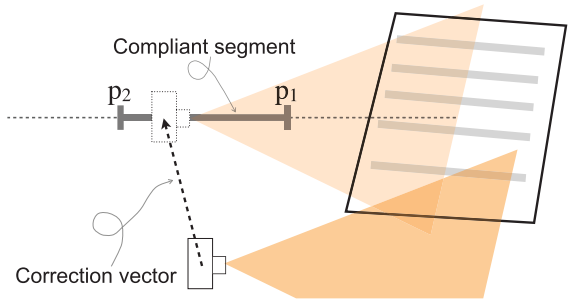


Fig. 5. The camera pose shown with the solid line is not compliant because part of the document is outside of the camera's field of view. If the camera is moved by the correction vector, it will reach a position in the compliant segment. If the orientation is kept constant, the new pose is compliant. Figure from Cutter and Manduchi (2015).

letter-size ($8.5'' \times 11''$) document was considered compliant for the sake of this study if (1) all four corners of the printable area were visible, where in our case the printable area had top and bottom margins of $1.5''$ and left and right margins of $0.5''$, and (2) a small letter placed anywhere in the printable area could be seen in the picture at enough resolution that it could be read accurately by OCR. A "small letter" could be, for example, a lowercase "x" character typed in 12-point Arial font, which has height of 4.23mm. By "accurately readable by OCR," we mean that the height of the letter in the image should be of at least 12 pixels (Zandifar and Chahine 2002). Note that compliance was defined only in geometric terms: factors such as bad illumination or blur certainly contribute to the quality of OCR reading but were not considered in this study.

To compute the camera pose from a picture of the printed fiducials, we used an open source software package, ArUco,[10] implemented with the OpenCV library. The ArUco fiducials were printed on a letter-size sheet in known locations (see Figure 4). The software was used to detect the location of the visible fiducials, and from this the camera pose. Only one fiducial is necessary for pose estimation, but accuracy is increased when multiple fiducials are seen. The software was able to process 20 images per second on average, although in practice the effective frame rate was smaller due to other concurrent processing on the phone.

Given the camera pose (computed with respect to a reference system centered at the document), the homography (perspective transformation (Hartley and Zisserman 2003)) mapping points in the paper sheet to pixels can be easily computed. This information was used to compute compliance of the current pose, based on the criteria discussed earlier (visibility of all corners of the document's printable area, minimum resolution).

For the guidance modality, we devised an algorithm[11] that produces a *correction vector* taking the camera to a *compliant pose* (a pose from which a compliant picture can be obtained) if camera orientation was kept constant. More precisely, the correction vector links the current camera position with the closest point in the *compliant segment* (shown in Figure 5), defined as the set of points on a line through the center of the sheet, parallel to the optical axis of the camera, such that each point in the segment represented a compliant camera location under the current orientation. The compliant segment for a given camera orientation is defined by two endpoints, $\mathbf{p}_1$ and $\mathbf{p}_2$, where $\mathbf{p}_2$ is higher (with respect the document) than $\mathbf{p}_1$. With the system used in our study (iPhone 4S), the heights of $\mathbf{p}_1$ and $\mathbf{p}_2$ were 28cm and 42cm, respectively. Note that if the slant of

[10] http://www.uco.es/investiga/grupos/ava/node/26.
[11] A simpler version of this algorithm was originally proposed in Cutter and Manduchi (2013).

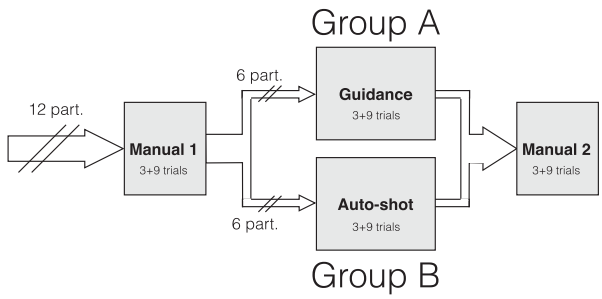


Fig. 6. Study 1 experimental setup.

the camera with respect to the sheet normal is larger than a threshold (noncompliant orientation), the compliant segment for the current camera orientation may contain no points, meaning that to reach a compliant pose, the camera needs to be reoriented.

Different types of information were produced by the system via synthetic speech. If a noncompliant orientation was detected during trials in the autoshot or guidance mode, the system uttered the sentence "Reset orientation" to prompt the user to reorient the phone, ideally bringing it parallel to the document. On detection of a compliant pose, the system uttered the sentence "Pose compliant," terminating the trial. In the guidance modality, instructions were produced by means of a short sentence containing directions along at most two Cartesian axes, and precisely those in need of the largest correction (e.g., "Move up 5 and forward 3" or "Move left 4"). Units were expressed in centimeters, and the reference system was fixed with respect to the paper sheet. We felt that specifying three vector coordinates (e.g., "Move up 5, forward 3 and left 8") would generate exceedingly long sentences and possibly become confusing.

4.1.3 Design. Figure 6 shows the different steps of the experiment. At the beginning, each participant was given a description of the functioning of the system. All participants first underwent a series of trials with the manual modality (*manual 1*). Then the six participants in Group A experimented with the guidance modality, whereas the remaining six (Group B) were tested with the autoshot modality. Finally, one more series of trials with the manual modality was conducted (*manual 2*). The purpose of this last step was to verify whether experience with an interactive modality (autoshot or guidance) could help our participants learn to take better pictures of the document even without system interaction. In other words, we wanted to find out whether feedback-rich modalities could be used for self-training on using a regular mobile OCR system in manual mode. Participants underwent 12 trials for each interaction modality and were informed that the first 3 trials of each session were to be considered practice trials (the results of these three initial trials were not included in the analysis).

At the beginning of each trial, the document was placed on a desktop surface at random orientation within ±45 degrees, and the iPhone running our app was placed flat on top of the bottom right corner of the document with its camera facing down. Participants were then asked to pick up the phone and move it to take a good snapshot of the document, with the interaction modality being tested for that trial. In the manual modality, participants were asked to take a snapshot by pressing either volume button located at the side of the iPhone 4s when they thought a readable picture could be taken. (At the beginning of the experiment, participants were advised that to take a compliant picture, they needed to hold the phone at a height of approximately 1 to 1-1/2 feet.) In the other modalities, participants were tasked with moving the phone until the system informed them that a compliant pose was reached. In the guidance modality, participants were asked to follow the

directions provided by the system and were informed that metric directions were expressed in centimeters. A time-out period of 150 seconds was set for all trials; if a snapshot was not taken within the time-out period, the trial was terminated. During the experiment, participants were free to try whichever hand positions worked best for them. Several participants experimented with multiple positions of the phone-holding hand throughout the experiment. Most participants decided to sit for the duration of the experiment, although three participants chose to stand for all or part of the experiment. The whole experiment lasted 1 hour or less for each participant.

Proficiency at capturing text was measured in two ways. The first, and most important, metric was *readability R*, which was defined as the number of equivalent 12-point characters in the printable area that were OCR readable from the image divided by the total number of characters in the printable area, assuming the the printable area was filled with 12-point characters in a grid. (This grid was designed based on standard intercharacter and interline spacing.) Note that an image was considered compliant only when its readability R was equal to 1. The second measure taken was *time to completion* T_c: the time lapsed from the start of the trial until a hi-res snapshot was taken. (Timed-out trials were assigned $T_c = 150$ seconds.) A shorter time to completion is desirable, provided that the resulting snapshot is readable.

Both measures were taken for trials in all three modalities. However, it should be clear that these measures have different relevance depending on the modality. In particular, *time to completion* is mostly relevant for the autoshot and guidance modalities, where the decision to take a hi-res snapshot is made by the system. When testing the manual modality, the user decides when to take a snapshot without any feedback from the system; hence, in this case, time to completion is a purely subjective measure (the user could decide to take a snapshot after an arbitrarily short or long time). For what concerns readability, the snapshots taken by the system in the autoshot and the guidance modalities were considered to be perfectly OCR readable (as the image was deemed compliant). Hence, the readability measure is only of importance for the manual modality.

Two hypotheses were tested by this study:

- *Hypothesis 1:* Trials with the guidance modality should result in shorter time to completion T_c (on average) than trials with the autoshot modality.
- *Hypothesis 2:* Image readability R should be higher, on average, for trials in the *manual 2* set than in the *manual 1* set.

Hypothesis 1 is justified by observing that without guidance, blind users can rely solely on their spatial awareness to move the phone to a compliant pose. Guidance, if well designed, should help one reach a compliant pose faster. Hence, verification of Hypothesis 1 would be evidence that guidance mechanisms (which add a significant interaction component) could indeed improve proficiency at capturing OCR-readable text.

Hypothesis 2 formalizes our conjecture that experience with the autoshot or guidance modality could help one become more proficient at capturing a good image of the document. Note that in our experiments, the OCR output was never produced; hence, users received no feedback about whether the snapshot they took was OCR readable. This means that one could not possibly learn how to take a good shot from experimenting with the manual trials alone.

4.2 Results

4.2.1 Quantitative Results. The time to completion T_c results are shown, together with relevant statistics, in Figure 7(a) for the autoshot and guidance modalities. When testing Hypothesis 1, we took the logarithm of T_c, as this was shown to increase Gaussianity of the residuals of linear fit, based on visual inspection of the Q-Q plot.

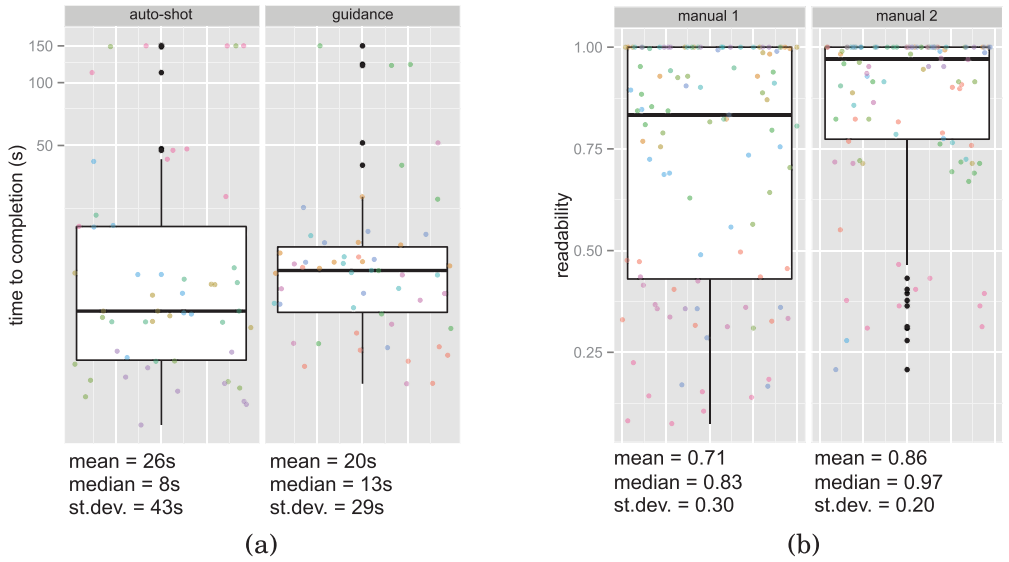


Fig. 7. Study 1. Times to completion (a) and readability (b) results. Each point corresponds to a trial, with the color characterizing the participant's ID. Points are randomly scattered on the *x*-axis for visualization.

Multiple sample repeated measures ANOVA analysis (Hedeker and Gibbons 2006) did not find a significant difference in the mean of $\log T_c$ between autoshot and guidance trials, and thus Hypothesis 1 was not confirmed. The larger variance of $\log T_c$ for autoshot can be appreciated visually from Figure 7(a); the F-test shows that the difference in variances is statistically significant (p = 2e-16).

We compared readability R values between *manual 1* and *manual 2* trials using a standard 2×2 mixed factorial design model. A significant difference in mean readability was found (p = 2e-3), with trials in the *manual 2* modality resulting in a larger mean value ($\bar{R} = 0.86$) than *manual 1* trials ($\bar{R} = 0.71$). Hypothesis 2 was thus confirmed. No interaction was found with the groups (A or B).

4.2.2 Failure Case Analysis. Analysis of individual results shows that although some participants were quite proficient at taking compliant pictures without system feedback (manual modality), others had serious difficulties. In particular, seven participants could not take a single compliant picture in the *manual 1* trials; three of them could not take any compliant picture in the *manual 2* trials either.

A failure (defined here as a snapshot taken in the manual mode that was not compliant) results from a noncompliant terminal camera pose (i.e., the pose of the camera at hi-res snapshot time). Some of these noncompliant terminal camera poses were hopelessly wrong, whereas others only needed a small adjustment to become compliant. Note that a noncompliant pose can always be made compliant by reorienting and repositioning the camera. In some cases, a simple reorientation while keeping the camera in place would be sufficient. In others, it would be sufficient to reposition the camera while keeping it in the same orientation. Some poses can be made compliant by either reorienting or repositioning the camera. By analyzing the terminal camera poses, we discovered that in 84% of the cases, a simple repositioning of the camera would have led to a compliant snapshot. In a smaller proportion of cases (59%), a compliant pose would have been reached by simply reorienting the phone. The more serious situation of a pose requiring both orientation and position adjustment occurred only 6% of the time.

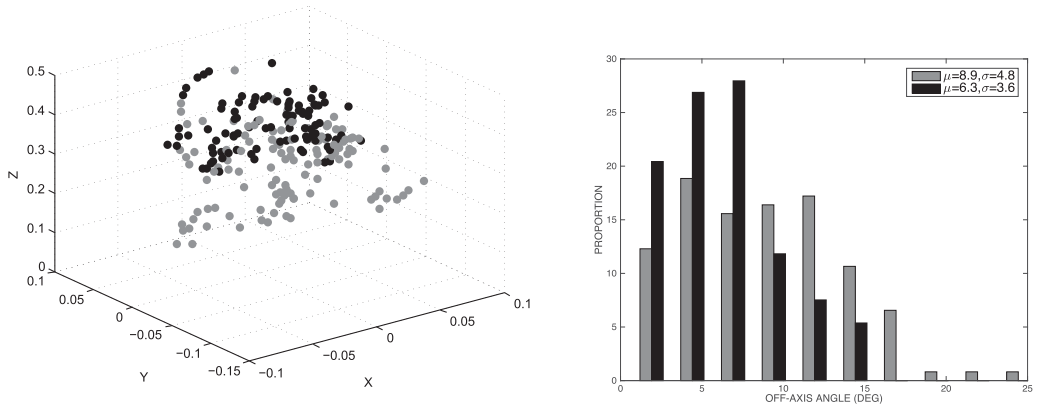


Fig. 8. Study 1. (a) 3D locations of terminal camera pose in the *manual* trials defined with respect to a reference system centered at the center of the paper sheet (units are in meters). Black represents a compliant pose. Gray represents a noncompliant pose. (b) Histogram of off-axis angles for compliant (black) and noncompliant (gray) terminal poses in the *manual* trials. Figures from Cutter and Manduchi (2015).

Figure 8(a) shows the location of the camera for terminal compliant poses (black dots) and terminal noncompliant poses (gray dots) in the manual modality. (Remember from Section 4.1.2 that locations higher than 42cm and lower than 28cm from the document were noncompliant.) The plot suggests that in many cases, noncompliance was due to participants keeping the phone too close to the document (the difference in height means between compliant and noncompliant poses was significant at p <1e-3.)

Figure 8(b) shows the histogram of off-axis angles (where the off-axis angle was defined as the angle between the camera's optical axis and the normal to the document) for terminal poses. Note that the off-axis angle, by itself, does not determine compliance: if the camera is located to the side of the document, a moderately large off-axis angle may be required for compliance. This histogram shows that on average, noncompliant poses were characterized by a larger off-axis angle than compliant poses (the difference in means was significant at p <1e-3.)

4.2.3 Qualitative Observations and Feedback. Several participants found the action of pressing a volume button to take a snapshot in the manual mode somewhat difficult to execute, especially if holding the phone with one hand, whereas others found it very natural. Two participants expressed concern about the possibility that while reaching with a finger for these buttons the phone may be inadvertently moved, generating blur or resulting in the picture taken from an incorrect location.

Two participants in Group A lamented the fact that guidance directions were issued in centimeters, a unit to which they were not accustomed. Note that we chose centimeters (rather than inches) so that commands could be issued as integer numbers with good enough resolution.

One participant in Group A strongly disliked the guidance modality. The median time to completion for this participant in the guidance trials was 48 seconds, which was much larger than the overall median time of 13 seconds. The same participant was unable to reach a compliant pose within the time-out period for three of the nine trials.

At the end of the experiment, each participant was asked to complete a short survey. Participants were asked to comment on several statements using a five-point Likert scale. The statements, reported verbatim in Table 1 along with the median responses, differed slightly across the two participant groups.

Table 1. Study 1 Survey

Questions for Group A (1 = Strongly Disagree; 5 = Strongly Agree)	Median Response
It was easy to follow the directions from the system.	5
The directions from the system helped me take better pictures of the document.	4
I feel that, after interacting with the system, I am now able to take better pictures of the document by myself.	4
If the guidance system were available as an app, I would be interested in using it.	5

Questions for Group B (1 = Strongly Disagree; 5 = Strongly Agree)	Median Response
The system helped me take better pictures of the document.	4
I feel that, after interacting with the system, I am now able to take better pictures of the document by myself.	4
If this system were available as an app, I would be interested in using it.	5

4.3 Discussion

Several interesting observations can be drawn from the results of Study 1. Figure 7(b) shows that our participants exhibited a wide diversity of proficiencies at taking compliant snapshots without help from the system. By observing the participants during the experiment, it was clear that some were much more "methodical" than others in the way they moved the phone to take a snapshot. Interestingly, as shown by Figure 8(a), participants tended to take snapshots at a short distance from the document: the maximum recorded height of a snapshot was 44cm, which is only slightly above the maximum compliant height (42cm). As mentioned earlier, participants were informed that the correct height was approximately between 1 and 1-1/2 feet, but it seems that they preferred to err on the lower end. Of course, since no feedback was provided in the manual trials, participants did not have a means to correct what could be a biased perception of the camera height. Interestingly, this tendency did not change even after experience with the autoshot and guidance trials, in which participants had a chance to experiment firsthand the range of compliant heights.

The fact that Hypothesis 2 was confirmed seems to indicate that at least some of our participants learned the proprioception skills that are necessary to correctly position a camera. For example, a participant in Group A, after several trials with the guidance modality, said: "Aha now I've got it!" Similar "aha" moments occurred for other participants during autoshot or guidance, at which point the time to complete each trial dropped.

Perhaps the biggest surprise in Study 1 was the discovery that Hypothesis 1 was not confirmed—the average time to completion for the guidance modality was not significantly shorter than for autoshot. We believe that there may be two main reasons for this. The first reason, which has to do with the definition of compliance itself, will be discussed later in Section 5.3, in light of the data acquired in Study 2. The second reason is related to two aspects of the user interface, as discussed in the following.

Lack of explicit orientation guidance. As shown in Figure 8, noncomplaint images were often associated with excessive off-axis angles. Our guidance system gave directions in terms of translation but not of orientation; this was a deliberate choice to keep the complexity of directions low. Participants were advised to keep the iPhone horizontal; only on detection of a large off-axis angle was a synthetic speech warning produced. However, most participants found it difficult to reorient the phone correctly (horizontally), resulting in the off-axis warning being reissued several times

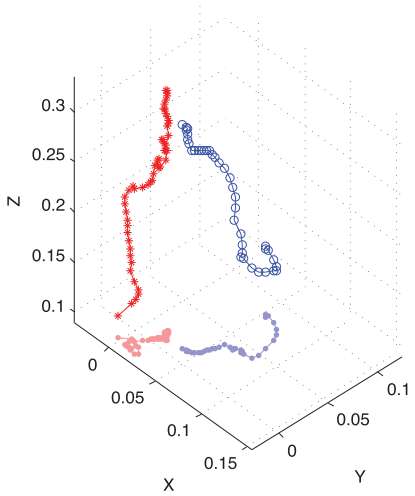


Fig. 9. The path taken by the camera location during two Study 1 trials, using the autoshot modality (red) and the guidance modality (blue). Units are in meters. The projection of the paths on the horizontal plane is shown with faded color. Circular blue marks and red asterisks are placed at constant time periods of 0.1 seconds. Only the portion of the path after a certain time lag is shown (as measurements cannot be taken when the camera is too close to the document.) This lag was 6.8 seconds for the path marked in red and 3.9 seconds for the path marked in blue.

before the orientation of the iPhone could be properly adjusted. When this happened, the whole process was slowed down, which generated frustration among some participants. This suggests that some form of orientation correction guidance could be beneficial. Indeed, as discussed earlier, in 59% of the noncompliant snapshot cases, a simple camera reorientation would have been sufficient to make the pose compliant, and in 16% of the cases this correction would in fact have been necessary.

Possibly disruptive guidance interface. The synthetic speech directions produced by the system contained precise metric indication of where to move the phone next. Ideally, the user would move the phone exactly as directed, ending up at a compliant pose. In fact, this was rarely the case. This resulted in participants in Group A following a discrete sequence of movements; after each movement, they would pause and wait for the system to produce the next direction. Part of the reason a compliant pose could not be reached in just one correction could be that participants could not make good use of the metric information provided with the instruction. In contrast, participants in Group B, who did not receive guidance, moved the phone of continuous motion; this allowed for a larger portion of space to be explored in the same amount of time. The difference in behavior for the two modalities can be noticed in Figure 9. The path marked in blue (guidance) is characterized by nonuniform velocity and several abrupt turns in response to a system instruction, whereas the path marked in red (autoshot) shows a more fluid motion.

5 STUDY 2

Study 2 was conducted in spring 2015 with nine blind participants. This experiment used documents printed with actual text rather than with fiducials. A specially designed computer vision algorithm processed the images taken by the camera to estimate compliance and produce guidance

directions. Hi-res snapshots were processed by state-of-the-art OCR software (ABBYY FineReader), and a measure of readability was defined based on the OCR output. Two different documents were used: one ("small font") printed in 10-point font, and one ("large font") printed in 16-point font. Both documents, containing a restaurant menu, were printed with black ink on a letter-size ($8.5'' \times 11''$) paper sheet. Figures 2(a) and (c) show the large-font document. This apparatus enabled more realistic tests for the different interaction modalities than in Study 1. However, this system did not allow for precise analysis of camera pose (and thus of the user motion), as made possible by the apparatus of Study 1. A lighter user interface, requiring less information processing by the user, was implemented in this study in hopes that it would result in smoother camera trajectories than observed in Study 1 and thus in more efficient exploration.

5.1 Method

5.1.1 Participants. We recruited nine participants for this study (four female and five male). Their ages ranged from 25 to 67 years, with a median age of 60. All participants were blind, except for at most some residual light perception. They were randomly assigned to two groups (A and B), such that Group A had five participants and Group B had four participants.

Seven of these participants had already participated in Study 1. Study 2 was conducted 2 years after Study 1, and therefore the risk of carryover effects (Doncaster and Davey 2007) was very small.

All participants owned a smartphone, although one of the participants only used her iPhone to make phone calls. Two participants also had a hearing impairment but were still able to hear instructions from the phone. Four of the nine participants had previous experience with mobile OCR applications (Text Detective, Prizmo, or KNFB Reader). The most commonly cited use case involved physical mail, including determining to whom a letter was addressed. By coincidence, all four participants with prior mobile OCR experience were assigned to Group A.

All participants described situations in their daily lives in which they desired to, but could not, access printed text. Examples included handouts distributed in class or at conferences, restaurant menus posted on a wall, and yoga schedules. Five of the participants owned a flatbed scanner that could be used for OCR. Their opinion was that mobile OCR may be preferable due to better ease to use, and also because they found that most flatbed scanner OCR software is obsolete.

5.1.2 Apparatus. We designed an app, implemented on an iPhone 6, which continuously processed the images acquired by the camera. When taking a high-res snapshot (as triggered by the user in the manual modality by pressing either of the two volume buttons placed on the side of the phone, or by the system in the autoshot or guidance modality), the flash was activated; this reduced exposure time (and thus motion blur) and increased the signal-to-noise ratio of the resulting image.

Similarly to Study 1, we defined a document image to be compliant if it had enough resolution to be OCR readable and if all text in the document was visible. In general, OCR can be assumed to work well when the height of the smallest characters is of 12 pixels (Section 4.1.2); however, based on the results of preliminary experiments, we decided to take a more conservative approach and require that the x-height be at least 18 pixels. Note that we ran OCR on hi-res images ($3{,}264 \times 2{,}448$ pixels), whereas compliance was computed on lower-resolution images (640×480 pixels, approximately five times less resolution in each direction). Hence, for an image to be considered compliant (based on lower-resolution mode analysis), the median x-height must be at least 4 pixels.

For what concerns the second compliance criterion (all text in the document visible in the image), we made the simplifying assumption that the document had some white padding around the text and required that this white padding be visible in a compliant image. We chose the width of

required padding to be at least three times the median *x*-height (see Figure 13(c)); note that this amount is larger than the distance between two text lines in typical documents, which reduces the risk that some lines of text could be cropped out in an image declared to be compliant. Of course, it is possible that consecutive paragraphs in the text may be separated by a space wider than our minimum required padding; in this case, our system may end up triggering an image capture of an individual paragraph rather than one of the whole document. In practice, we checked for the white padding violation by extending each segment of the rectangular bounding box of each text line (computed using the algorithm presented in the Appendix) by four times the median *x*-height, and by verifying that the new endpoints were contained in the image. If any text line did not have the required padding, the image was declared not compliant. To reduce the risk of false detections due to errors in the line grouping phase, we required that at least seven consecutive frames met the white padding and *x*-height constraints before triggering hi-res image capture in the autoshot and guidance modalities. Our system ran at 15 to 20 frames per second in most cases, although when the image was fully occupied by text, the frame rate reduced to 10 frames per second.

This fairly sophisticated definition of image compliance has, in our opinion, several advantages with respect to similar implementations in existing software products. For example, as mentioned earlier, Text Detective (which declares compliance as soon as some text is visible in the image) considers an image to be compliant even when part of the text is truncated (see Figure 1(b)). This undesirable result is avoided by our system thanks to the white padding condition. Compliance definition based on the detection of edge quadrilateral (Prizmo and KNFB Reader) does not guarantee that text is present or readable in the document (see Figure 3(b) and (c)) and fails if the document's edges are not clearly visible due to low contrast against a white background (see Figure 3(a)) or to occlusion (see Figure 2(c)). All of these situations are well managed by our algorithm. Of course, handling more complex situations with multiple columns, graphics, and text outside of the main printout area (e.g., footer and header) would require more complex compliance detection mechanisms.

With the guidance modality enabled, our system produces instructions (in the form of synthetic speech) guiding the user to move the phone to a position from which a compliant image of the document could be captured. Intuitively, if the *x*-height criterion is violated (median *x*-height less than 4 pixels), the image has insufficient resolution and the user needs to move the camera closer to the document (system utters the word "Lower"). If the white padding condition is violated in, say, the left side of the image, the camera needs to be moved to the left ("Left"). We allowed for two horizontal directions to be produced in the same sentence (e.g., "Backward left"). If the camera is too close to the document, possibly resulting in two or more sides with no visible white padding, the camera should be raised ("Raise"). Compared to Study 1, the directions produced by the new interface were shorter and did not contain quantitative metric information. As discussed in Section 4.3, we felt, based on observations of the Study 1 trials, that metric information was not well processed by our participants, and that adding it to the spoken instructions could actually slow down the exploration process.

Once the hi-res capture process was triggered, the phone generated a short melody that lasted for 1 second, after which it uttered the word "Wait," activated the flash, and took the snapshot. The purpose of this phase was to encourage the user to hold the phone still while the snapshot was taken to reduce the risk of motion blur. While the melody was being played back, data from the phone's accelerometer was analyzed and low-res images were continuously acquired and analyzed for compliance. If at some point a noncompliant frame was detected, or acceleration with magnitude larger than 0.05g was measured (meaning that the phone was moving, possibly resulting in a blurry picture), the hi-res image acquisition process was aborted.

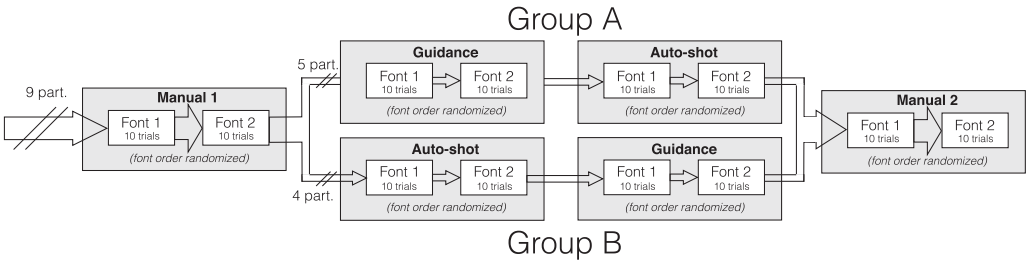


Fig. 10. Study 2 experimental setup.

To ensure good image quality (and avoid perspective distortion consequent to large slant), our system required that the phone be kept as horizontal as possible. The roll and pitch angles from the phone's accelerometer were measured at all times; if either of these angles was larger than 7 degrees, the phone produced a warning sound. This sound had different frequency depending on whether the roll or the pitch threshold was exceeded (if both angles exceeded the threshold, both sounds were produced). If the roll/pitch condition was violated during the hi-res capture process, the process was aborted.

Measurements taken in Study 2 were similar to those in Study 1 (time to completion T_c and readability R). However, as a specific OCR software (ABBYY FineReader) was used to process the hi-res snapshots, we were able to derive a measure of readability directly from the output of OCR. Specifically, readability R was defined as the ratio of the number of characters that were correctly recognized to the number of characters in the whole document. Both Hypotheses 1 and 2, defined in Section 4.1.3, were tested in this study.

5.1.3 Design. The experiment proceeded in a similar fashion to Study 1, with some important differences (see Figure 10):

(1) All nine participants experimented with both modalities, albeit in different order. The five participants in Group A started with the guidance modality, whereas the remaining four (Group B) started with the autoshot modality.
(2) Each series of 10 trials in each modality was divided into two consecutive batches of five trials each, if the same font size was used for the trials in the same batch. The order of font size for the two batches was randomized; the participants were not made aware of which font was used at each trial.
(3) Participants were allowed to experiment with the system before the beginning of each interaction modality session until they felt confident in its use.

At each trial, the paper sheet was placed in front of the participant, always in the same orientation (unlike Study 1). We also decided to increase the time-out period to 180 seconds. This made almost no difference, as in only one trial we recorded a time to completion between 150 and 180 seconds.

5.2 Results

5.2.1 Quantitative Results. The time to completion T_c results are shown, together with relevant statistics, in Figure 11(a) for the autoshot and guidance modalities. As in the case of Study 1, we took the logarithm of T_c when testing Hypothesis 1, as this was shown to increase Gaussianity of the linear fit residuals based on visual inspection of the Q-Q plot. Repeated measures ANOVA found a significant difference in the mean value of T_c between the two modalities (autoshot and guidance, p = 9e-9), thus confirming Hypothesis 1. Significant interaction was also found between

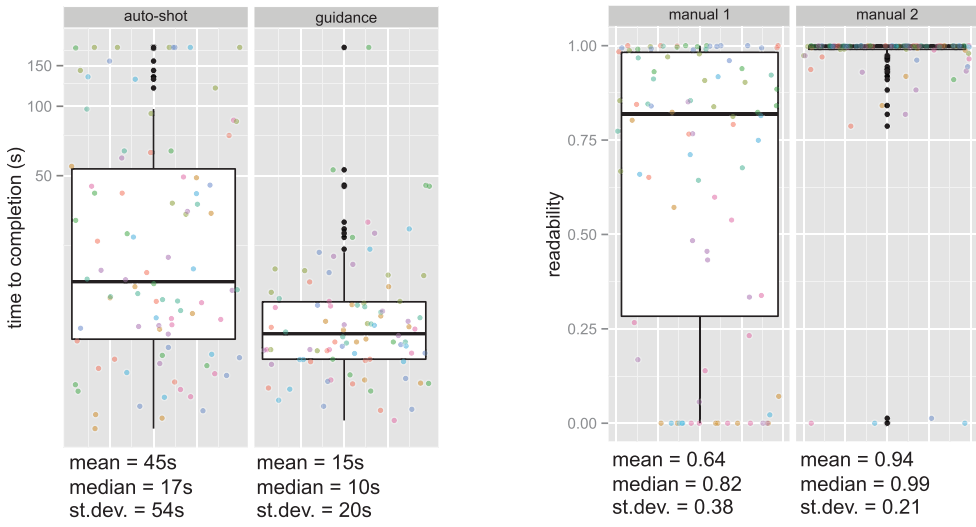


Fig. 11. Study 2. Times to completion (a) and readability (b) results. Each point corresponds to a trial, with the color characterizing the participant's ID. Points are randomly scattered on the *x*-axis for visualization.

Table 2. Relevant Statistics for Time to Completion T_c Measurements in Study 2

	Autoshot	Guidance
Group A	Mean = 53s Median = 18s st.dev. = 60s	Mean = 12s Median = 10s st.dev. = 5s
Group B	Mean = 34s Median = 17s St. dev. = 45s	Mean = 19s Median = 10s St. dev. = 28s

	Small Font	Large Font
Group B	Mean = 38s Median = 21s St. dev. = 47s	Mean = 15s Median = 9s st.dev. = 21s

Note: The left side shows the interaction between modality (autoshot vs. guidance) and Group (A vs. B). The right side shows the interaction between font size (small vs. large) and group (A vs. B).

the group (A or B) and modality, and between the font size and group. When analyzing the interaction, it was found that the modality was a significant factor for both levels of group, but the group was not a significant factor for either level of modality. In addition, the font size was a significant factor only for group B, whereas the group was not a significant factor for either font size. The relevant statistics of T_c for these significant interactions are shown in Table 2. Caution should be used when interpreting these interaction statistics due to the likely low power resulting from the small sample size (four or five participants in each group). The difference in variance of T_c between modalities (autoshot and guidance), which is well noticeable from Figure 11(a), was confirmed by the F-test (p = 2e-16).

Readability R was compared between *manual 1* and *manual 2* trials using repeated measures ANOVA. A significant difference in readability means was found between the two cases (shown in Figure 7(b); p = 4e-13), thus confirming Hypothesis 2. Significant interaction was found between modality (*manual 1* vs. *manual 2*) and group (A vs. B). Analysis of this interaction revealed that the modality was a significant factor for each group level (A or B), but the group was not a significant factor for either level of modality. The relevant statistics are summarized in Table 3.

Table 3. Relevant Statistics for Readability Measurements *R* in Study 2 Describing the Interaction Between Modality (*Manual 1* vs. *Manual 2*) and Group (A vs. B)

	Manual 1	*Manual 2*
Group A	Mean = 0.77 Median = 0.90 St. dev. = 0.31	Mean = 0.94 Median = 0.99 St. dev. = 0.21
Group B	Mean = 0.48 Median = 0.53 St. dev. = 0.41	Mean = 0.93 Median = 0.99 St. dev. = 0.22

Fig. 12. Two participants in Study 2 while operating the system.

The mean number of instructions given by the system per trial in the guidance modality was 4.7 (min = 0; max = 26; σ = 3.8; median = 4). Intervention order (Group A or B) was not shown to have a significant effect on the number of instructions. No correlation was found between the number of instructions given and the time to completion.

We also computed the proportion of time in each trial with the system giving an acoustic warning to signal that the phone needed reorienting. This was computed by dividing the number of frames with the warning activated by the total number of frames in the trial. The mean proportion was 0.09 (min = 0; max = 0.54; σ = 0.09; median = 0.07).

The mean number of hi-res image acquisition abort events per trial (due to excessive acceleration or loss of compliance) was 1.72, with a standard deviation of 5.63. The distribution of these events was highly skewed by one participant, who experienced 7.4 abort events per trial on average, with a few trials affected by a large number of such events (up to 50 per trial). On analysis of the video collected during these trials, it appears that the cause for this anomalous behavior was a combination of system malfunctioning (frequently generating incorrect text line segmentation resulting in incorrect compliance detection) and user behavior (the participant kept the phone still while the acquisition would cyclically start, only to be aborted shortly afterward). After removing data from this participant, the mean number of acquisition abort events reduced to 1.01, with a standard deviation of 1.95.

5.2.2 Qualitative Observations and Feedback. Each participant developed his or her own personal strategy for moving the phone and aiming at the document. Six participants held the phone with one hand, whereas the remaining three used two hands (Figure 12). One participant stood during the *manual 1* and *manual 2* trials, and remained seated for the other trials. All other participants conducted all trials from a seated position.

Several participants (particularly those who had prior mobile OCR experience) would feel the edges of the document to help themselves center and orient the camera correctly. For example, one participant described his strategy as follows: "Feel top edge, then bottom edge, then point at left top." Camera alignment was certainly important. For example, one participant in most trials kept the phone misaligned with the text lines by 30 degrees or more, which made compliant acquisition difficult. She appeared to be aware of this problem and frequently tried to reset the phone orientation by placing it back to the start position halfway through the trial. Another participant moved the phone very quickly to probe various areas. One participant with prior mobile OCR experience felt for the camera and positioned it at the center of the document.

Centering and orienting the camera with the document is necessary but not sufficient to obtain a compliant picture. For example, one participant kept the camera too close to the document for successful reading in the *manual 1* trials, which led to very poor performance. Another participant had difficulty reaching a compliant pose in the autoshot phase (with trials lasting 60 seconds or more), until he found a successful strategy resulting in compliant image acquisition in short time. He described his strategy as gently rocking the phone back and forth while raising it slowly.

Three participants suggested that the system should allow the user to "modulate" the amount of guidance given. One of them said this: "Sometimes people need a lot of correction at the beginning, but other times I want it to just let me know if it is a good picture at the end."

One participant had several issues with the guidance modality. He complained about the lag between instructions and sometimes did not trust the instructions issued by the system (on different occasions, he remarked "I don't believe you" and "I am dubious"). Likewise, another participant said that he did not believe the "Raise" or "Lower" instructions given but found the horizontal positioning instructions sometimes helpful.

One of the participants remarked that continuous feedback from the guidance modality was easier to interact with than the KNFB Reader's "Field of view report" (described in Section 3). As he put it, "You have to think about what to do next. Plus you have to hold it in place. Sometimes I rotate it the wrong way." He preferred how our interface did not require the user to request information but instead provided instructions continuously. In fact, another participant complained that sometimes directions in the guidance modality were not produced frequently enough.

One participant said this: "There are a lot of apps out there but there are not a lot that give instructions. It's a great app. It's so frustrating to scan a document." She described both interactive modalities as "frustration savers . . . it might take more time to line up but it will at least save time ultimately because you don't have to take four to five pictures."

At the end of the experiment, participants were asked to answer a short set of questions about their experience. The survey questions and median response are reported in Table 4.

5.3 Discussion

The results from Study 2 validated our Hypothesis 1: the guidance modality appeared to enable faster execution, with a reduction of the average time to completion by 33% with respect to the autoshot modality. Participants also perceived the guidance modality as easier with respect to the other modalities (see Table 4). Part of the reason for the discrepancy between the Study 1 and 2 results (data from Study 1 did not validate Hypothesis 1) may lie in the new, lighter interface implemented in Study 2, which may have helped the participants reach a compliant pose faster by following directions from the system. Unfortunately, unlike the system in Study 1, the Study 2 apparatus could not track the camera pose, so we cannot directly compare the camera trajectories in the two cases. Another possible explanation is that the compliance detection algorithm used in Study 2 could have been more conservative than the one in Study 1, meaning that it declared compliance (thus triggering a hi-res snapshot) only when the phone reached a subset of all possible

Table 4. Study 2 Survey

Survey Questions (1 = Strongly Disagree; 5 = Strongly Agree)	**Median Response**
I believe the pictures I took without feedback at the start of the experiment were completely readable.	1
I believe the pictures I took without feedback at the end of the experiment were completely readable.	2
The directions from the system helped me take better pictures of the document.	4

Perceived Difficulty (1 = Very Difficult; 5 = Very Easy)	**Median Response**
Manual modality	3
Autoshot modality	3
Guidance modality	4

compliant poses. Reaching a pose in this smaller subset by pure chance could take longer than when guided by the system. This seems to be confirmed indirectly by the fact that reaching a compliant pose in the autoshot mode took longer on average (45 seconds) in Study 2 than in Study 1 (26 seconds).

Our Hypothesis 2 was also confirmed: our participants were more proficient at taking compliant pictures with no feedback from the system after experiencing feedback-rich interaction modalities. In the *manual 2* trials, the average readability was 94%, compared to 64% for the *manual 1* trials. We should caution the reader, however, that, as observed earlier, it is difficult to translate these readability values into a measure of practical utility. The fact that 6% of the characters in the text were incorrectly read may mean different things when these characters formed, say, the last two rows of the document (in which case all remaining text is clearly understandable) than when a whole set of columns is missing (because it is outside of the camera's field of view), in which case interpreting the text content may be challenging.

Interestingly, the font size was not found to be a significant factor in time to completion or readability. To understand this result, it is useful to consider the viewing geometry and its effect on the image resolution and framing. To take a picture with the iPhone 6 camera framing the whole text width, which was kept constant for the small- and large-font documents, and assuming that the camera is kept horizontal and well centered with the document, the camera should be held at a distance no smaller than 30cm from the document. The maximum distance is determined by the font size and by the resolution requirements for correct OCR reading (Section 5.1.2). In our case, the maximum distance was 38cm for the small-font document and 63cm for the large-font document. The fact that successfully accessing the two document types required a similar amount of effort suggests that a main reason for failure (noncompliant snapshots) could be that the phone was kept at too short of a distance from the document (the minimum reading distance being the same for both documents). This would be consistent with the results of Study 1 (Section 4.2.2), which showed that in many cases failure was due to the camera being kept too close to the document.

6 GENERAL DISCUSSION

We summarize in the following some general considerations from the two studies.

Unsupervised training is possible. In both experiments, the ability of our participants to take OCR-readable images without system feedback improved significantly after experiencing with a

feedback-rich modality. The exact mental process that goes into this improvement is not clear. Participants might have simply developed some "muscle memory" from the trials in the autoshot or guidance modality, they might have calibrated their perception of the correct distance to the document, and/or they might have devised some means to keep the phone centered with respect to the document. In any case, it is remarkable that through a completely unsupervised process (the experimenter never provided assistance), participants learned to take better pictures of the document. Interestingly, participants did not seem to fully realize it: the median response to the second survey question shown in Table 4 (which asked whether, in the participants' opinion, the snapshots taken in the *manual 2* phase were readable) was only slightly more positive ("disagree" vs. "strongly disagree") than the median response to the same question referred to the *manual 1* trials.

This improvement could not have been possible through repeated manual trials due to the lack of feedback from the system. It should be noted, however, that had the participants been given a chance to hear the outcome of OCR at each snapshot, they could conceivably have learned through trial and error to correctly position the phone—even without experience with real-time interaction from autoshot or guidance. Anecdotal evidence from discussion with our participants who experimented with existing OCR apps revealed that this trial-and-error process can be very time consuming.

System feedback is time efficient. Without system feedback, the likelihood of taking non-OCR readable snapshots is substantial, especially for untrained users (see Figures 7(b) and 11(b)). Considering that it took about 10 seconds on average for our participants to take a snapshot in the manual modality, the overall time required to take an OCR-readable picture may be large if multiple trials are necessary. Using the guidance mode, it took our participants only 20 seconds (Study 1) to 15 seconds (Study 2) on average to acquire an OCR-readable shot of a printed letter-size sheet.

Guidance is effective, but not always. Although it may seem logical that following instructions from the system should allow one to reach a compliant pose faster, this may not always be the case (as shown by the Study 1 trials). We have advanced two possible justifications for this somewhat counterintuitive phenomenon. One reason could be that processing relatively long, detailed system feedback may actually slow down the process of moving the camera in search of a compliant pose. The fact that lengths were expressed in centimeters, a unit to which our participants may not have been accustomed, might have increased the cognitive load. The other reason may be connected with the ability of the system to detect all compliant poses. Users of a very conservative system that triggers a snapshot only when the camera reaches a location in a small subset of possible poses may benefit greatly from following directions from the system. If the space of compliant poses that can be correctly detected by the system is large, one such pose could be reached by chance, by simply moving the camera over the document; in this case, guidance from the system may be unnecessary.

7 CONCLUSIONS

We have presented the results of two experiments meant to evaluate whether and how feedback from a computer vision system could help a blind user of a mobile OCR app take OCR-readable pictures faster. The two experiments used very different systems, yet they were organized in a similar fashion, and similar measurements were collected in both. The system used in Study 1 did not process the images with OCR but instead tracked the camera pose in real time by means of special AR fiducials and indirectly inferred the readability of text from a certain camera pose via geometric reasoning. It afforded analysis of camera trajectories and measurements such as the distance of the camera to the document and its off-axis orientation. The system used in Study 2

used a fast text spotter and text line detector to measure in real time whether an image of the document was OCR readable. Both systems also provided a guidance modality that gave directions to the user about where to move the camera to increase the likelihood to capture an OCR-readable picture. These directions were produced via synthetic speech, with the system in Study 1 creating richer directions than the one in Study 2. Both studies measured the average time to completion for trials with the interactive modalities (autoshot and guidance) and text readability for trials with the manual modality. Results showed that in the case of Study 2, the guidance modality led to a reduction of time to completion with respect to the autoshot modality. In both studies, the readability of pictures of the document taken with the manual modality were higher for trials taken after experience with the autoshot and/or guidance modalities.

From a practical viewpoint, this work has shown that current mobile OCR systems, which at best support a very simplified guidance modality, have room for improvement. However, to be really effective, a guidance mechanism requires a fairly sophisticated computer vision module for compliance detection, which must work at a high frame rate and possibly support multiple document layouts (which may include different font size, pictures insets, the presence of header and footers, etc.). In addition, our work has highlighted the importance of a carefully designed user interface. Relatively minor details, such as the length of the sentences uttered by the system or the modality chosen to warn the user that the phone is incorrectly tilted, may affect the time involved in taking an OCR-readable image.

We should note that this work has focused on one particular problem of operating a mobile OCR system without sight, which is the acquisition of a well-framed image at an acceptable resolution. Other issues affecting the quality of OCR reading include bad illumination, glare, cast shadows, and motion blur. Future work will consider extensions of our interaction modalities to deal with these additional nuisance factors.

APPENDIX

A ALGORITHMS FOR TEXT SPOTTING AND LINE GROUPING

This appendix describes the computer vision algorithm that is at the core of the compliance detection strategy of Study 2. This algorithm is divided into two components: a fast text spotting module, followed by an oriented line grouping algorithm.

Text spotting techniques have received increasing attention by the computer vision community. Unlike OCR, text spotters do not (usually) decode text; rather, they are specialized in the fast detection (and localization) of any text content in the image. The text spotter utilized in this contribution builds on the popular stroke width transform (SWT) algorithm by Epshtein et al. (2010). Compared to other more recent techniques based on convolutional deep networks (Zhang et al. 2016; Bissacco et al. 2013; Qin and Manduchi 2016), this algorithm enables very fast processing even on a smartphone.

We would like to emphasize that our intent in this work was simply to design a system that worked properly for our purposes. Other text spotters may outperform our algorithm (we did not run a comparative analysis); however, to the best of our knowledge, no other system was demonstrated that computes the visible white padding in the image in addition to spotting text.

A.1 Connected Component Segmentation

The first step of the algorithm is the detection of connected components from the SWT, an algorithm for the detection of text strokes based on the observation that text strokes have approximately constant width and tend to form a connected graph within each character. Following the original SWT algorithm (Epshtein et al. 2010), we first compute an edge map using the work of

Canny (1986). We then cast a ray from each edge pixel in the direction of the local gradient. A ray is accepted if it intersects another edge point with the opposite gradient direction (within a tolerance of ±30 degrees). Intuitively, accepted rays are those with a good likelihood to section a character stroke. The length of each accepted ray is measured and recorded at each pixel intersected by the ray, resulting in a stroke width map. A graph is formed on this map, where two pixels are connected by an edge if they are neighbors in the pixel grid, and if the larger recorded stroke width of the two is less than three times the smaller one.

A.2 Letter Classification

In the original SWT algorithm, connected components were classified as text characters based on certain geometric properties (aspect ratio, height, stroke width variance, and the number of encapsulated connected components). A simple classifier was designed by defining thresholds for these parameters; this classifier was trained on the ICDAR training dataset (Karatzas et al. 2013). To increase classification robustness, we considered more features: Euler number (number of holes), perimeter to area ratio, number of horizontal crossings, stroke width variance, and stroke width over height of the connected component (the first three inspired by the work of Neumann and Matas (2012)). We used a random forest classifier (Breiman 2001) on this feature with 100 trees and maximum depth of 5. The training data for this classifier had positive samples selected from the connected components that overlapped the ground truth regions and negative samples mined from the wrong predictions using the original SWT algorithm (in other words, a negative sample was a connected component that was incorrectly classified as "character" by SWT). Similarly to Neumann and Matas (2012), we tuned the classifier to favor recall over precision: a connected component is classified as a character if this is the prediction of at least 25 trees. In the following, we will use the term *character* to define a connected component that has been classified as such by our algorithm.

A.3 Generating Candidate Document Orientations

The next step is to determine text lines. We assume that text lines are mutually parallel in the image, although this is not strictly true even when the text lines are parallel in the document (due to perspective deformation). Our strategy is to first identify a number of candidate line orientations, which are then validated by associating characters to lines and checking for consistency.

We start from the edge map, which was computed earlier as part of SWT analysis. (To ensure real-time processing, the edge map is subsampled by 8 in each direction.) We then search for dominant lines using the Hough transform (Duda and Hart 1972). Only lines that have length larger than one-eighth of the the longest side of the input image are kept. We used k-means to cluster the set of line orientations, where for each line we added an orthogonal orientation to the set. This was inspired by the observation that, when applied to Latin script document images, the Hough transform predominately hallucinates lines that are either aligned or orthogonal to the actual document text lines (Figure 13(a) and (b)). We considered $k = 5$ clusters; however, if two resulting orientations were within 2 degrees each other, we only kept one of the two. The resulting set of candidate orientation is denoted by Θ.

A.4 Selecting the Best Text Line Orientation

Before assigning characters to text lines, we computed the median x-height across characters. (Note: In typography, x-height commonly refers to the distance between the baseline and mean line of lowercase letters in a typeface.) Specifically, our estimate of the x-height was given by the median value of the set formed by the lengths of the smaller side of the bounding boxes of all characters. For each candidate orientation θ in Θ, we grouped characters into text lines after first

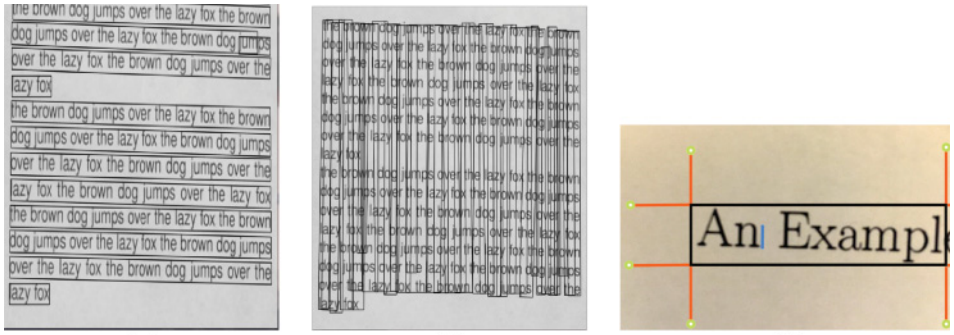


Fig. 13. (a, b) Two possible text line hypotheses after initial rectifying rotation. Notice that the horizontal hypothesis has no intersecting text lines, whereas the vertical hypothesis has many. (c) The black rectangle represents a text line bounding box. The orange segments (with length equal to three times the estimated *x*-height, shown by the blue segment) are used to test the white padding condition (see Section 5.1.2). Note that the left side of the text line satisfies the white padding condition, but not the right side.

rotating the image around its center by angle θ. Intuitively, if this is the correct text line orientation, we expect all characters in a line to share the same y-coordinate; the distribution of y-coordinates of all characters in the image should have multiple modes, one per line. Based on this observation, we computed the histogram of the y-coordinates of the centroids of all characters and selected the largest peak; this is expected to correspond to the most densely populated horizontal line. All characters whose centroid had a y-coordinate differing from the location of the histogram peak by no more than the x-height were associated to this line and removed from the corpus of characters. Then the histogram was computed again on the remaining characters, iterating until no more than two characters could be assigned to the horizontal line defined by the histogram peak. For each line, we computed the minimum bounding box containing all of its characters, with sides pairwise parallel to the sides of the (rotated) image.

At this point, we had a set of text lines for each orientation, along with the lines' bounding boxes. To select the "correct" orientation θ (and associated text lines), we computed a metric motivated by two observations: (1) the correct orientation should create lines that contain most of the detected characters, and (2) the bounding boxes of the lines should be well separated (see Figure 13). We translated these observation into an empirical metric that is the linear combination of two terms: (1) the proportion of characters that are associated with a text line and (2) the inverse of the proportion of text lines that overlap at least another text line. The orientation that produced the highest value for this metric, along with the associated lines' bounding boxes, was produced in output. This information was then used to determine compliance, as explained in Section 5.1.2.

ACKNOWLEDGMENTS

The authors would like to thank all participants in the experiments for their willingness to help, their patience, and for providing useful feedback and comments.

REFERENCES

Hend S. Al-Khalifa. 2008. Utilizing QR code and mobile phones for blinds and visually impaired people. In *Proceedings of the International Conference on Computers for Handicapped Persons*. 1065–1069.

Jeffrey Bigham, Chandrika Jayant, Andrew Miller, Brandyn White, and Tom Yeh. 2010b. VizWiz::LocateIt—enabling blind people to locate objects in their environment. In *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'10)*. DOI: http://dx.doi.org/10.1109/CVPRW.2010.5543821

Jeffrey P. Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C. Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samuel White, and Tom Yeh. 2010a. VizWiz: Nearly real-time answers to visual questions. In *Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology (UIST'10)*. ACM, New York, NY, 333–342. DOI: http://dx.doi.org/10.1145/1866029.1866080

Alessandro Bissacco, Mark Cummins, Yuval Netzer, and Hartmut Neven. 2013. PhotoOCR: Reading text in uncontrolled conditions. In *Proceedings of the IEEE International Conference on Computer Vision*. 785–792.

Erin Brady, Meredith Ringel Morris, Yu Zhong, Samuel White, and Jeffrey P. Bigham. 2013. Visual challenges in the everyday lives of blind people. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'13)*. 2117–2126. DOI: http://dx.doi.org/10.1145/2470654.2481291

Leo Breiman. 2001. Random forests. *Machine Learning* 45, 1, 5–32.

Rickey Dale Burks, Charles Lee Oakes III, Randy Ray Morlen, Bharat Prasad, Michael Frank Morris, and Xia Hua. 2012. Systems and methods to use a digital camera to remotely deposit a negotiable instrument. US Patent 8,290,237.

John Canny. 1986. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 8, 6, 679–698. DOI: http://dx.doi.org/10.1109/TPAMI.1986.4767851

James Coughlan and Roberto Manduchi. 2013. Camera-based access to visual information. In *Assistive Technology for Blindness and Low Vision*, R. Manduchi and S. Kurniawan (Eds.). CRC Press, Boca Raton, FL, 219–246.

Michael P. Cutter and Roberto Manduchi. 2013. Real time camera phone guidance for compliant document image acquisition without sight. In *Proceedings of the 2013 12th International Conference on Document Analysis and Recognition*. IEEE, Los Alamitos, CA, 408–412.

Michael P. Cutter and Roberto Manduchi. 2015. Towards mobile OCR: How to take a good picture of a document without sight. In *Proceedings of the 2015 ACM Symposium on Document Engineering*. ACM, New York, NY, 75–84.

C. Patrick Doncaster and Andrew J. H. Davey. 2007. *Analysis of Variance and Covariance: How to Choose and Construct Models for the Life Sciences*. Cambridge University Press.

Richard O. Duda and Peter E. Hart. 1972. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM* 15, 1, 11–15. DOI: http://dx.doi.org/10.1145/361237.361242

B. Epshtein, E. Ofek, and Y. Wexler. 2010. Detecting text in natural scenes with stroke width transform. In *Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'10)*. DOI: http://dx.doi.org/10.1109/CVPR.2010.5540041

Richard Hartley and Andrew Zisserman. 2003. *Multiple View Geometry in Computer Vision*. Cambridge University Press.

Donald Hedeker and Robert D. Gibbons. 2006. *Longitudinal Data Analysis*. Vol. 451. John Wiley & Sons.

Bill Holton. 2016. A day in the life: Technology that assists a visually impaired person throughout the day. *AFB AccessWorld Magazine* 17, 2. Available at http://www.afb.org/afbpress/pubnew.asp?DocID=aw170202.

Chandrika Jayant, Hanjie Ji, Samuel White, and Jeffrey P. Bigham. 2011. Supporting blind photography. In *Proceedings of the 13th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS'11)*. 203–210. DOI: http://dx.doi.org/10.1145/2049536.2049573

Shaun K. Kane, Brian Frey, and Jacob O. Wobbrock. 2013. Access lens: A gesture-based screen reader for real-world documents. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'13)*. ACM, New York, NY, 347–350. DOI: http://dx.doi.org/10.1145/2470654.2470704

Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, Masakazu Iwamura, Lluis Gomez i Bigorda, Sergi Robles Mestre, Joan Mas, David Fernandez Mota, Jon Almazan Almazan, and Lluis Pere de las Heras. 2013. ICDAR 2013 Robust Reading Competition. In *Proceedings of the 2013 12th International Conference on Document Analysis and Recognition (ICDAR'13)*. DOI: http://dx.doi.org/10.1109/ICDAR.2013.221

Roberto Manduchi and James M. Coughlan. 2014. The last meter: Blind visual guidance to a target. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, 3113–3122.

Lukáš Neumann and Jiří Matas. 2012. Real-time scene text localization and recognition. In *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'12)*. IEEE, Los Alamitos, CA, 3538–3545.

Siyang Qin and Roberto Manduchi. 2016. A fast and robust text spotter. In *Proceedings of the 2016 IEEE Winter Conference on Applications of Computer Vision (WACV'16)*. IEEE, Los Alamitos, CA, 1–8.

Roy Shilkrot, Jochen Huber, Wong Meng Ee, Pattie Maes, and Suranga Nanayakkara. 2015. Fingerreader: A wearable device to explore printed text on the go. In *Proceedings of the 33rd Annual Conference on Human Factors in Computing Systems (CHI'15)*. 2363–2372.

Lee Stearns, Ruofei Du, Uran Oh, Catherine Jou, Leah Findlater, David A. Ross, and Jon E. Froehlich. 2016. Evaluating haptic and auditory directional guidance to assist blind people in reading printed text using finger-mounted cameras. *ACM Transactions on Accessible Computing* 9, 1, 1.

Deborah Stein. 1998. The Optacon: Past, Present, and Future. Retrieved July 5, 2017, from https://nfb.org//Images/nfb/Publications/bm/bm98/bm980506.htm.

Ender Tekin and James M. Coughlan. 2010. A mobile phone application enabling visually impaired users to find and read product barcodes. In *Proceedings of the 12th International Conference on Computers Helping People With Special Needs (ICCHP'10)*. 290–295. DOI: http://dl.acm.org/citation.cfm?id=1880751.1880800

Marynel Vázquez and Aaron Steinfeld. 2012. Helping visually impaired users properly aim a camera. In *Proceedings of the 14th International ACM SIGACCESS Conference on Computers and Assessibility (ASSETS'12)*. 95–102. DOI: http://dx.doi.org/10.1145/2384916.2384934

Ali Zandifar and Antoine Chahine. 2002. A video based interface to textual information for the visually impaired. In *Proceedings of the 4th IEEE International Conference on Multimodal Interfaces (ICMI'02)*. 325. DOI: http://dx.doi.org/10.1109/ICMI.2002.1167016

Zheng Zhang, Chengquan Zhang, Wei Shen, Cong Yao, Wenig Liu, and Xiang Bai. 2016. Multi-oriented text detection with fully convolutional networks. arXiv:1604.04018.

Yu Zhong, Pierre J. Garrigues, and Jeffrey P. Bigham. 2013. Real time object scanning using a mobile phone and cloud-based visual search engine. In *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS'13)*. Article No. 20. DOI: http://dx.doi.org/10.1145/2513383.2513443

Yu Zhong, Walter S. Lasecki, Erin Brady, and Jeffrey P. Bigham. 2015. RegionSpeak: Quick comprehensive spatial descriptions of complex images for blind users. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI'15)*. 2353–2362.

Received September 2016; revised January 2017; accepted March 2017